**1 Crea una tabla relacionada con el esquema que has cargado denominada CAMPAÑAS\_PUBLICITARIAS, la cual tendrá un *identificador (clave)* numérico sin decimales, un *título*, el *identificador de la categoría* de producto, y un valor *descuento\_adicional* de tipo numérico con dos decimales. Existirá un regla que chequee que el campo *descuento\_adicional* es un valor entre 0 y 1.**

CREATE TABLE campañas\_publicitarias (

id int primary key,

titulo varchar2(70),

id\_categoria int,

descuento\_adicional number (4,2),

constraint campa\_publi\_fk\_categoria foreign key (id\_categoria) references categorias,

constraint valida\_desc check(descuento\_adicional>0 and descuento\_adicional<1)

);

**2 Nombre de los empleados que NO han Atendido a Samuel Abadia. Hay que utilizar exclusivamente subconsultas.**

select id\_empleado, nombre, apellidos

from empleados

where id\_empleado not in (select id\_empleado

from pedidos

where id\_cliente = (select id\_cliente

from clientes

where nombre ='Samuel' and apellidos='Abadia'));

ID\_EMPLEADO NOMBRE APELLIDOS

----------- ------------------------- -------------------------

709 Domingo Dumbo

**3 Nombre y ciudad de los clientes que no han comprado productos de la categoría ' Bicicletas'.**

select nombre, ciudad

from clientes

where id\_cliente not in (select id\_cliente

from pedidos natural join detalles\_pedidos natural join productos

where id\_categoria = (select id\_categoria

from categorias

where descripcion = 'Bicicletas'));

NOMBRE CIUDAD

------------------------- ------------------------------

Susana Robledo

German Acebeda (La)

Daniel Santa Ana

Camilo Ladrillar

Javier Robledo

**4 Identificador de los empleados, su nombre e importe obtenido por cada empleado de las ventas de productos de la categoría Accesorios y de los productos cuyo nombre es “StaDry Cycling culote”.**

select id\_empleado, nombre, sum(cantidad\*precio) as total

from detalles\_pedidos natural join pedidos natural join empleados

where numero\_producto in

((select numero\_producto

from productos

where nombre ='StaDry Cycling culote')

union

(select numero\_producto

from productos join categorias using (id\_categoria)

where categorias.descripcion ='Accesorios'))

group by id\_empleado, nombre;

**También**

select id\_empleado, nombre, sum(cantidad\*precio) as total

from detalles\_pedidos natural join pedidos natural join empleados

where numero\_producto in

((select numero\_producto

from productos join categorias using (id\_categoria)

where nombre ='StaDry Cycling culote' or categorias.descripcion ='Accesorios'))

group by id\_empleado, nombre;

ID\_EMPLEADO NOMBRE TOTAL

----------- ------------------------- ----------

707 Catalina 58431.01

704 Carol 46580.1

708 Susana 47596.38

705 Pedro 51769.3

701 Ana 44774.77

706 David 41426.96

703 Mateo 43783.67

702 Maria 55177.56

8 filas seleccionadas

**5 Identificador de empleado, nombre e importe obtenido por el/los empleado/s que más pedidos ha/n realizado.**

select id\_empleado, nombre, sum(precio\_total)

from empleados natural join pedidos

where id\_empleado in (select id\_empleado

from pedidos

group by id\_empleado

having count(\*)= (select max(count(\*))

from pedidos

group by id\_empleado))

group by id\_empleado, nombre;

ID\_EMPLEADO NOMBRE SUM(PRECIO\_TOTAL)

----------- ------------------------- -----------------

707 Catalina 736127.67

Solo medio punto, podría haber más de uno y aquí solo saca 1

select id\_empleado, nombre, sum(precio\_total)

from empleados natural join pedidos

where id\_empleado = (select id\_empleado

from (select count(\*) as total, id\_empleado

from pedidos

group by id\_empleado

order by total desc)

where rownum=1)

group by id\_empleado, nombre;

**6 Obtener el nombre de los empleados y código postal de los clientes a los que hayan vendido en algún pedido, productos con precio por unidad comprendido entre 500 y 1900 €.**

select distinct empleados.nombre, empleados.apellidos, clientes.cod\_postal

from empleados join pedidos using (id\_empleado) join clientes using (id\_cliente) join detalles\_pedidos using (numero\_pedido)

where precio between 500 and 1900;

135 filas seleccionadas

**7 Obtener todos los datos de los 10 pedidos más recientes hechos por clientes de las provincias de CACERES ó MADRID cuyo nombre empieza por 'A'.**

select \* from (

select pedidos.\*

from pedidos join clientes on pedidos.id\_cliente=clientes.id\_cliente

where provincia in ('CACERES', 'MADRID') and nombre like 'A%'

order by pedidos.fecha\_pedido DESC)

where rownum <11;

NUMERO\_PEDIDO FECHA\_PEDIDO FECHA\_ENVIO ID\_CLIENTE ID\_EMPLEADO PRECIO\_TOTAL

------------- ------------ ----------- ---------- ----------- ------------

938 28/02/08 28/02/08 1011 705 6463.54

923 26/02/08 29/02/08 1009 707 1229.3

904 22/02/08 26/03/08 1009 708 7673

895 21/02/08 21/02/08 1011 705 16062.35

899 21/02/08 25/02/08 1009 705 5071.7

885 20/02/08 23/02/08 1009 701 173

883 19/02/08 22/02/08 1011 704 7646

816 08/02/08 11/02/08 1011 701 0

805 05/02/08 05/02/08 1009 702 7493.85

787 01/02/08 03/02/08 1009 707 13876.7

10 filas seleccionadas

**8 Obtener el identificador, nombre del empleado y número de clientes atendidos por el empleado que menos clientes distintos ha atendido.**

select id\_empleado, nombre, apellidos, count(\*) as clientes\_atendidos

from (select distinct id\_empleado, id\_cliente

from pedidos) natural join empleados

group by id\_empleado, nombre, apellidos

having count(\*) = (select min(count(\*))

from (select distinct id\_empleado, id\_cliente

from pedidos)

group by id\_empleado);

ID\_EMPLEADO NOMBRE APELLIDOS CLIENTES\_ATENDIDOS

----------- -------------------- ------------------ ------------------

703 Mateo Berges 24

Medio punto

select id\_empleado, nombre, apellidos, count(\*) as clientes\_atendidos

from pedidos natural join empleados

group by id\_empleado, nombre, apellidos

having count(\*) = (select min(count(\*))

from pedidos

group by id\_empleado);

ID\_EMPLEADO NOMBRE APELLIDOS CLIENTES\_ATENDIDOS

----------- -------------------- ------------------ ------------------

703 Mateo Berges 105

(no son clientes distintos, solo hay 28 clientes)

**9 Historifica los pedidos cuya fecha de envío sea anterior a “1.1.2008” (cópialos en las tablas correspondientes con nombre \_HISTORICO y bórralos de las tablas originales).**

insert into pedidos\_historico

select \* from pedidos where fecha\_envio < to\_date('01.01.2008');

**589 filas insertadas.**

insert into detalles\_pedidos\_historico

select \* from detalles\_pedidos where numero\_pedido in (select numero\_pedido from pedidos\_historico);

**2.483 filas insertadas.**

delete from detalles\_pedidos where numero\_pedido in (select numero\_pedido from pedidos\_historico);

**2.483 filas eliminado**

delete from pedidos where numero\_pedido in (select numero\_pedido from pedidos\_historico);

**589 filas eliminado**

**En este orden; si no da errores.**